Personal Privacy through Understanding and Action: Five Pitfalls for Designers

Scott Lederer, Jason Hong, Anind Dey, and James Landay

IRB-TR-03-035

September, 2003

Intel Research Berkeley

Personal Privacy through Understanding and Action: Five Pitfalls for Designers

Scott Lederer, Jason Hong
EECS, UC Berkeley
Berkeley, CA USA
{lederer,jasonh}@cs.Berkeley.edu

Anind K. Dey
Intel Research, Berkeley
Berkeley, CA USA
anind@intel-research.net

James Landay
Intel Research, Seattle
Seattle, WA USA
landay@intel-research.net

ABSTRACT
People create and maintain personal privacy by *understanding* the privacy implications relevant to a situation and influencing them through intuitive social *action*. It is a challenge for designers of interactive systems to empower these human-level processes of understanding and action through the limited technical mechanisms of feedback and control. To help meet this challenge, we present five pitfalls to avoid when designing interactive systems with personal privacy implications, on or off the desktop. These pitfalls are: obscuring potential information flow, obscuring actual information flow, emphasizing configuration over action, lacking coarse-grained control, and inhibiting social nuance. These pitfalls are based on the literature, on analyses of existing privacy-affecting systems, and on our own experiences designing a user interface for managing privacy in ubiquitous computing. We illustrate how some existing research and commercial systems—our prototype included—fall into these pitfalls, and how some avoid them. Designs that avoid them provide feedback and control mechanisms optimized to support the understanding and action required to create and maintain personal privacy.

Author Keywords
Privacy, Interaction Design, Ubiquitous Computing

ACM Classification Keywords
H.5.2 [Information Interfaces and Presentation]: User Interfaces – Theory and methods, User-centered design; K.4.1 [Public Policy Issues] – Privacy

INTRODUCTION
Westin defined information privacy as "the claim of individuals, groups or institutions to determine for themselves when, how and to what extent information about them is communicated to others" [37]. Though normally associated with the protection of personal data collected by *institutions*, this definition also aligns with *interpersonal* privacy, whereby people intuitively negotiate the many social tensions of everyday life [31]. Herein we collapse both notions into *personal privacy*: the processes by which an individual selectively discloses personal information—*e.g.*, shopping history or location—to institutions and to other people.

There has been a tremendous amount of research into the needs, wants, and realities of personal privacy in the context of technical systems. For example, many polls have shown considerable public concerns about privacy on the Internet [10, 34, 35]. There have also been interviews and surveys exploring privacy design issues for context-aware systems [17, 21, 24]; studies exposing privacy perceptions and practices in groupware [30], multimedia environments [2], and location-aware systems [4]; and experiments revealing usability problems affecting privacy in email [38] and file-sharing [15] applications. Despite this abundance of research and design knowledge, many systems still make it hard for people to maintain privacy on and off the desktop. We claim this is because designs fail to empower the human processes of *understanding* and *action*.

People create and maintain personal privacy in non-technical settings by understanding the privacy implications relevant to a situation and intuitively acting to influence them. The involvement of technical systems complicates this process, because systems have to support understanding and action through the limited feedback and control mechanisms of the relevant device or application. Too often, these technical mechanisms do not adequately support these human processes. Designs should optimize these mechanisms to empower informed social action through the system. This paper can help designers achieve that goal.

The contribution of this paper is a set of five pitfalls to avoid when designing for personal privacy on or off the desktop, based on an analysis of several privacy-affecting systems. Avoiding these pitfalls results in feedback and control mechanisms that empower the understanding and action people need to manage personal privacy. Many systems that fall into these pitfalls have encountered privacy controversies (*e.g.*, browsers), while those that avoid them enjoy considerable commercial or social success (*e.g.*, instant messaging).

Although some of our pitfalls may appear obvious, many systems continue to fall into them. Indeed, we fell into them

Submitted to CHI 2004 (#379)

ourselves in developing a user interface prototype for managing personal privacy in ubiquitous computing environments [25]. Despite the input of our formative interviews, surveys, and literature review, an evaluation indicated a series of fundamental missteps in our design rationale. Further analysis uncovered evidence of similar missteps in many existing commercial and research systems. We decided to investigate the misconceptions that led to these common design flaws.

Our five pitfalls are the outcome of this investigation. We have compiled them to serve as a guide for designers of privacy-affecting systems, who should carefully avoid them throughout the design cycle. Naturally, not all of the pitfalls will apply to every system; they should be interpreted within the context of the system being designed. Avoiding them will help create feedback and control mechanisms that empower users to create and maintain personal privacy through understanding the relevant privacy implications and intuitively acting to influence them.

A Preview of the Five Pitfalls

We have clustered our five pitfalls into those that primarily affect users' *understanding* of a system's privacy implications and those that primarily affect their *action* through the system. In practice, these two processes are intertwined, but we believe categorizing our pitfalls this way can help designers understand and remember them.

Understanding

1. *Obscuring potential information flow*. Designs should not obscure the nature and extent of a system's *potential* for disclosure. Users can make informed use of a system only when they understand its capabilities and the proper scope of its privacy implications.
2. *Obscuring actual information flow*. Designs should not conceal the *actual* disclosure of information through a system. Users should understand exactly what information is being disclosed to whom.

Action

3. *Emphasizing configuration over action*. Designs should not require excessive configuration to create and maintain privacy. They should enable users to manage privacy as a natural consequence of their primary actions involving the system.
4. *Lacking coarse-grained control*. Designs should not fail to provide an obvious, top-level mechanism for halting and resuming the disclosure of personal information.
5. *Inhibiting social nuance*. Designs should not inhibit users from transferring established social practices to emerging technological contexts.

The rest of this paper is organized as follows. First, we briefly discuss the design and evaluation of *Faces*, our UI prototype for managing personal privacy in ubiquitous computing settings. The negative results of the evaluation motivated our investigation into the design missteps encoded in our five pitfalls. We then describe the five pitfalls, with illustrative examples from both our own and related work. We then discuss the pitfalls' implications for the design process, including an extension of Norman's elucidation of mental models. Finally, we offer negative and positive case studies of systems that, respectively, fall into and avoid the pitfalls.

FACES: (MIS)MANAGING UBICOMP PRIVACY

Our investigation into the pitfalls began after we encountered them firsthand while designing *Faces*, a prototype for specifying privacy preferences in ubiquitous computing (ubicomp) environments. Ubicomp envisions computation embedded throughout everyday environments to support arbitrary human activities [36]. But by distributing and concealing displays and sensors, it complicates interaction [5]. This can leave users unaware of or unable to intentionally influence the disclosure of personal information—such as location and identity—as they go about their activities in augmented environments. To address this, we designed Faces to (1) provide feedback about information disclosures in a log, not unlike a financial transaction statement, and (2) support the specification of disclosure preferences, such as *who* can obtain *what* information *when*. Users would employ feedback in the log to iteratively refine their disclosure preferences over time.

As we will show below, the design of Faces involved some crucial missteps, which, as we discovered, are also present in other systems. What clued us in to the fundamental nature of these missteps is that we made them despite a substantive requirements gathering effort (details in [25]). We reviewed the literature. We interviewed twelve citizens, walking them through a series of scenarios to elicit how they might think about privacy in ubicomp. We surveyed 130 people on the web to investigate factors that determine privacy preferences in ubicomp [24]. And we iterated through a series of low-fidelity designs. The upshot of our findings was that the identity of the inquirer is a primary determinant of users' privacy preferences, but the situation in which the information is disclosed is also important.

Accordingly, we designed Faces to let users assign different disclosure preferences to different *inquirers*, optionally

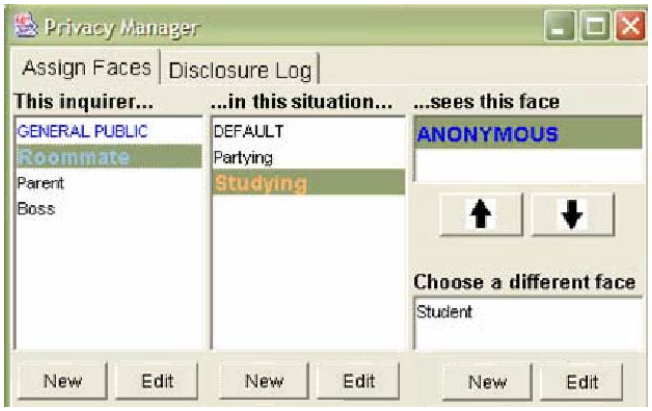

Figure 1. UI for creating and assigning faces. Each face holds precision preferences for blurring information disclosed to an given inquirer when the user is in a given situation.

parameterized by *situation* (a conjunction of location, activity, time, and nearby people). We employed the metaphor of *faces* to represent disclosure preferences. This is a fairly direct translation of Goffman, who posited that a person works to present himself to an audience in such a way as to maintain a consistent impression of his role in relation to that audience [14]. Users specify their preferences by creating 3-tuples of *inquirers*, *situations*, and *faces*, with each 3-tuple meaning "if *this* inquirer wants information about me when I'm in *this* situation, show her *this* face" (Figure 1). Wildcards are allowed in the inquirer and situation slots to handle requests involving inquirers or situations that the user has not specified.

Each face alters the information disclosed to the inquirer by specifying the *precision* at which to disclose it. Faces supports four levels of precision, from Undisclosed (disclose nothing) to Precise (disclose everything). Each face contains a precision preference for each of the following information dimensions: identity, location, activity, and information about nearby people. Adjusting the precision of information—in effect, blurring it—can desensitize it, allowing for different versions of the same information to reach different inquirers, depending on the situation [24]. In doing so, Faces operationalizes three of Adams and Sasse's four factors that determine the perception of privacy in sensed environments: recipient, context, and sensitivity [3]. We did not directly address the fourth factor, usage, because it is often impractical to predict an observer's information usage [6].

A formative evaluation revealed fundamental problems with our design. We asked five participants to use the system to configure their privacy preferences regarding two inquirers and two situations of their choice. We then described a series of hypothetical but realistic scenarios involving those same inquirers and situations and asked the participants what information, if any, they would prefer to disclose in those scenarios. It turned out that the participants' configured preferences often differed from their stated preferences during the scenarios. Further, they had trouble remembering the precision preferences inside their faces, clouding their ability to predict disclosure. Finally, they expressed discomfort with the indirection between faces and the situations in which they apply. In their minds, a situation and the face one "wears" in it are inseparable. These results together illustrate the misstep of separating privacy management from the contexts in which it applies.

In sum, while Faces modeled Goffman's theory *in* the interface, it inhibited users from *practicing* identity management *through* the interface. Users had to think explicitly about privacy in the abstract—and instruct the system how to shape it—instead of managing it intuitively through their actions *in situ* [31]. Faces does little to empower real-time *understanding* of the privacy implications of an augmented environment and it does little to support the creation of privacy through intuitive *action* therein. After realizing the misconceptions in our seemingly reasonable design process, we found evidence of similar mistakes in other systems. Further analysis culminated in our five pitfalls to avoid when designing for personal privacy, presented below with evidence of other designs both falling into and avoiding them.

FIVE PITFALLS IN DESIGNING FOR PRIVACY

Our pitfalls encode an analysis of common problems in interaction design across several systems, constituting a preventative guide to help designers avoid mistakes that may appear obvious but are still being made.

They fit into a history of analyses and guidelines on developing privacy-sensitive systems. Bellotti and Sellen argue the importance of feedback and control to the privacy process [6]. But Palen and Dourish argue for conceiving of privacy not as an access control problem, but as an ongoing boundary negotiation process; they suggest *genres of disclosure* as a sort of design pattern approach to supporting it [31]. Our five pitfalls are, in part, an effort to reconcile Palen and Dourish's theoretical insights about how people maintain privacy with Bellotti and Sellen's practical guidelines for designing feedback and control to support it. In reaching for this middle ground, we have tried to honor the fair information practices, as promoted by Langheinrich [22], and to minimize information asymmetry between users and observes, as argued by Jiang *et al*. [20].

Concerning Understanding: Two Pitfalls

Our first two pitfalls primarily involve the user's *understanding* of the system's privacy implications. Designs can enable this understanding by illuminating (1) the system's potential for information disclosure and (2) the actual disclosures made through it.

Pitfall 1: Obscuring Potential Information Flow

Designs need to make clear the nature and extent of the system's *potential* for disclosure. Designs that neglect this can give false impressions about the ways in which they affect personal privacy. Users can make informed use of a system only when they understand its capabilities and the proper scope of its privacy implications. This means understanding what kinds of information the system conveys and what kinds of observers it conveys it to. Exposing this information is crucial for empowering users to predict the social consequences of using the system.

Designs should clarify the kinds of information the system can disclose. This includes identifiable *personae* (*e.g.*, true name, email addresses, credit card numbers, social security number) and *activities* (*e.g.*, locations, purchases, web browsing histories, communications, A/V records, social networks). People cannot maintain consistent identities without knowing which of their activities can be associated with which of their personae [23].

Designs should also clarify what kinds of observers can obtain the user's personal information through the system. Some systems only involve interpersonal disclosure

(revealing sensitive information to another person), some only institutional (companies or governments), and some both. Designs should clarify the involvement of each, making clear the extent to which primarily interpersonal disclosures (*e.g.*, chat) involve incidental institutional disclosures (*e.g.*, workplace chat monitoring) and, conversely, the extent to which primarily institutional disclosures (*e.g.*, workplace cameras) involve secondary interpersonal disclosures (*e.g.*, mediaspaces).

"Privacy" is a broad term whose unqualified use as a label can mislead users into thinking a system protects or erodes privacy in ways it does not. Making the scope of a system's privacy implications clear will help users understand its capabilities and limits. This in turn provides grounding for comprehending the actual flow of information through the system, addressed in the next pitfall.

Evidence: Falling into the Pitfall. An easy way to obscure a system's privacy scope is to present its functionality ambiguously. One example is Microsoft's Windows operating systems. The Windows Internet control panel offers ordinal degrees of privacy protection (from Low to High) for Internet use. The functional meaning of this scale is unclear to average users and, as it turns out, this mechanism does not affect general Internet use through the operating system; its scope is limited to a particular web browser's cookie management heuristics. Similarly, Anonymizer.com's free anonymizing software can give the impression that all Internet activity is anonymous when the service is active, but in actuality it only affects web browsing, not email, chat, or other services. A for-pay version covers those services.

Another example is found in Beckwith's report of an eldercare facility using worn transponder badges to monitor the locations of residents and staff [4]. Many residents perceived the badge only as a call-button (which it was) but not as a persistent location tracker (which it also was). They did not understand the scope of its privacy implications.

Similarly, some hospitals use badges to track the location of nurses for efficiency and accountability purposes, but they can neglect to clarify what kind of information the system conveys. One concerned nurse wrote, erroneously, "They've placed it in the nurses' lounge and kitchen. Somebody can click it on and listen to the conversation. You don't need a Big Brother overlooking your shoulder" [32].

Evidence: Avoiding the Pitfall. Many web sites that require an email address for creating an account give clear notice on their sign-up forms that they do not share email addresses with third parties or use them for extraneous communication with the user. Clear, concise statements like these help clarify scope.

Another successful design is Tribe.net, a social networking service that carefully conveys that members' information will be made available only to other members within a certain number of degrees of social separation.

Pitfall 2: Obscuring Actual Information Flow

Having addressed the user's need to understand a system's *potential* privacy implications, we move now to the issue of *actual* instances of disclosure. Designs need to make clear the actual disclosure of information through the system. Users should understand exactly what information is being disclosed to whom. The disclosure should be obvious to the user as it occurs; if this is impractical, notice thereof should be available as soon as is reasonable. There should be just enough feedback to inform but not overwhelm the user.

By avoiding both this and the prior pitfall, designs can clarify the extent to which users' actions engage the system's range of privacy implications. This enables users to understand the consequences of their use of the system thus far, and it empowers them to predict the consequences of future use. In the Discussion section, we will elaborate on how avoiding both of these pitfalls can support the user's mental model of his personal information flow.

Evidence: Falling into the Pitfall. Web browser support for cookies is a persistent example of obscuring information flow [27]. Most browsers do not, by default, indicate when a site sets a cookie or what information is disclosed through its use. The prevalence of third-party cookies and web bugs (tiny web page images that monitor who is reading the page) exacerbates users' ignorance of who is observing their browsing activities.

Another example of concealed information flow is in the Kazaa P2P file-sharing application, which has been shown to facilitate the concealed disclosure of highly sensitive personal information to unknown parties [15].

Another simple example is locator badges like those described in [4, 17], which generally do not inform their wearers about who is locating them.

Evidence: Avoiding the Pitfall. Friedman *et al's.* redesign of cookie management improves browsers' ability to show what information is being disclosed to what web sites [13].

Instant messaging systems often employ a symmetric design that informs the user when someone else wants to add her to his contact list, allowing her to do the same. By then letting users see and adjust their own status (*e.g.*, "Busy" or "Out to Lunch"), they inform users *who* can see *what* about them. This gives individuals a better understanding of how they are presenting themselves.

AT&T's mMode Find Friends service, which lets mobile phone users locate other users of the service, informs the user when someone else is locating them. They learn *who* is obtaining *what* (their location).

Concerning Action: Three Pitfalls

Our last three pitfalls primarily involve the user's *actions* involving the system. These pitfalls encode the recognition that fine-grained control of privacy emerges from the nuanced manipulation of coarse-grained controls. The details are in human actions, not technical parameters.

Pitfall 3: Emphasizing Configuration over Action

Designs should not require excessive configuration to create and maintain privacy. They should enable users to manage privacy as a natural consequence of their primary actions involving the system.

Palen and Dourish write, "setting explicit parameters and then requiring people to live by them simply does not work, and yet this is often what information technology requires… Instead, a fine and shifting line between privacy and publicity exists, and is dependent on social context, intention, and the fine-grained coordination between action and the disclosure of that action" [31]. But because configuration has become a universal UI design pattern, many systems fall into the pitfall of configuration.

Configured privacy breaks down for at least three reasons. First, in real settings, users manage privacy intuitively in the course of their primary actions; they do not spell out their privacy needs in an auxiliary, focused effort [38]. Configuration imposes an awkward requirement on users.

Second, the act of configuring preferences is too easily desituated from the contexts in which those preferences apply. Users are challenged to predict their needs under hypothetical circumstances, and they can forget their preferences over time. If they predict wrongly, or remember incorrectly, their configured preferences will differ from their *in situ* needs, creating the conditions for an invasion of privacy. We found evidence of this when evaluating Faces.

Third, users tend to avoid configuring systems anyway; default settings prevail [26, 30]. This means many users are unlikely to bother to manage their privacy if it needs to be configured. If users are to manage their privacy at all, it needs to be done in an intuitive fashion, as a predictable outcome of their situated actions involving the system.

People generally do not set out to explicitly protect their privacy; rather, they set out to do a task, with protecting their privacy as part of the overall context of the task. Designs should take care not to invert this process.

Evidence: Falling into the Pitfall. Many systems emphasize explicit configuration of privacy, including experimental online identity managers [7, 19], P2P file-sharing software [15], web browsers [27], email encryption software [38], and our Faces prototype.

Evidence: Avoiding the Pitfall. Successful solutions can involve some measure of configuration, but tend to embed it into the actions necessary to use the system. Web sites like Friendster.com and Tribe.net allow users to regulate information flow by modifying their social networks—a process that is embedded into the use of these applications.

Georgia Tech's In/Out Board [11] lets users reveal or conceal their presence in a workspace by badging into an entryway device. Its purpose is to convey this information, but it can be intuitively used to withhold information as well, by falsely signaling your in/out status.

Ignoring the moral implications, another example involves camera surveillance. When someone is aware of a camera's presence, she tends to intuitively adjust her behavior to present herself as she wants to be perceived [12].

Cadiz and Gupta propose a smart card that one could hand to a receptionist to grant limited access to her calendar to schedule an appointment; he would hand it back right afterwards. No one would have to fumble with setting permissions. They also suggest extending scheduling systems to automatically grant meeting partners access to a user's location in the minutes leading up to a meeting, so they can infer his arrival time. The action of scheduling a meeting implies limited approval of location disclosure [9].

Pitfall 4: Lacking Coarse-Grained Control

Designs should not fail to provide an obvious, top-level mechanism for halting and resuming the disclosure of personal information. Users are accustomed to turning a thing off when they want its operation to stop. Turning off information flow is an intuitive action that supports privacy.

Beyond binary control, a simple linear control may also be appropriate in some cases (*cf.*, audio devices' mute and volume controls). Ubicomp systems that convey location or other context could incorporate both a *precision dial* and a *hide button*, so users can either adjust the precision at which their context is disclosed or decidedly halt disclosure.

In the general case, users can become remarkably adept at wielding coarse-grained controls to yield nuanced results (*e.g.*, driving a car). Coarse-grained controls tend to reflect their state, providing direct feedback and freeing the user from having to remember whether she set a preference properly. This helps users accommodate the controls and even co-opt them in ways the designer may not have intended. Examples specific to privacy include: setting a door ajar, covering up or repositioning cameras [6, 18], turning off a phone or using its invisible mode rather than navigating its privacy-related options, and removing a locator badge.

While some fine-grained controls may be unavoidable, the flexibility that fine-grained controls are intended to provide is often neglected by users (see Pitfall #3). Flexibility in the control of privacy often comes not from within the system, but from the user's nuanced use of coarse-grained controls.

Evidence: Falling into the Pitfall. Many e-commerce web sites recommend to shoppers items that were purchased by other shoppers with similar shopping histories. While this is a useful service, there are times when a shopper does not want the item at hand to be included in his profile; he effectively wants to shop anonymously during the current session. Even though the merchant will know about the purchase, the shopper may not want his personalized shopping environment—which others can see—to reflect this private purchase. We have encountered no web sites that provide a simple mechanism for excluding the current purchase from our profiles.

Similarly, most web browsers still bury their privacy controls under two or three layers of configuration panels [27]. Third-party applications that expose cookie control have begun to appear (*e.g.*, GuideScope.com).

Further, wearable locator-badges like those described in [17] and [4] do not have power buttons. One could remove the badge and leave it somewhere else, but sometimes simply turning it off would be more practical or preferable.

Evidence: Avoiding the Pitfall. Systems that expose simple, obvious ways of halting and resuming disclosure include easily coverable cameras [6], mobile phone power buttons, chat systems with invisible modes, the In/Out Board [11], and our Faces prototype, with a button on a handheld application that overrides current settings.

Pitfall 5: Inhibiting Social Nuance

Designs should not inhibit existing social practices. People are accustomed to creating and maintaining privacy through a range of established, nuanced practices. Examples include plausible deniability (whereby the observer cannot determine whether a lack of disclosure was intentional) [28, 39], and disclosing ambiguous information. Designs that inhibit nuanced disclosure practices will constrain users from managing privacy intuitively through their action.

Technical systems involving multiple users are notoriously awkward at supporting social nuance [1], yet privacy is a highly nuanced social activity. This leaves designers of privacy-affecting systems in a dilemma. As a compromise, when systems are incapable of *actively supporting* nuance, they should at least *avoid constraining* it.

To an appropriate degree, designs should allow users to withhold information without fear of repercussion (*e.g.*, through plausible deniability) and to disclose ambiguous information (*e.g.*, imprecise location). These are some of the subtle ways in which people fine-tune their privacy.

Good designs also adapt to new practices that users develop around them. Key to this is not imposing excessive rigidity on users' behavior. A system that supports flexible usage practices allows users to exploit its subtleties in (hopefully positive) ways that its designer may have never intended [8, 16]. This process empowers users to *own* their use of the system and, hence, to manipulate it to create the subtle shades of privacy that social behavior requires.

Evidence: Falling into the Pitfall. Some researchers envision context-aware mobile phones that can inform the caller of the user's activity, to help explain why their call was not answered [33]. But this can prohibit users from exploiting plausible deniability. There can be value in keeping the caller ignorant of the reason for not answering.

Location-tracking systems like those described in [17] and [4] constrain users' ability to incorporate ambiguity into their location disclosures. Users can only convey a single precision of location or, at times, nothing at all.

Evidence: Avoiding the Pitfall. Mobile phones, push-to-talk phones [39], and instant messaging let users exploit plausible deniability by not responding to hails and not having to explain why.

Although privacy on the web is a common concern, a basic function of HTML allows users to practice ambiguous disclosure. Forms that let users enter false data facilitate anonymous account creation and service provision.

Tribe.net supports another subtle real-world practice. Tribe allows users to partition their social networks into "tribes," thereby letting pre-existing groups represent themselves online, situated within the greater networks to which they are connected. In contrast, Friendster.com users each have a single set of friends that cannot be functionally partitioned.

DISCUSSION

Having described the five pitfalls and provided evidence of systems that fall into and avoid them, we now examine some of the deeper implications they have for design. We begin by elaborating on the influence of our first two pitfalls on the user's mental model of his information disclosures. This leads to the introduction of a new conceptual tool to help the design process. Then we present an analytical argument for why designs that avoid our five pitfalls can support the human processes of understanding and action necessary for personal privacy maintenance. Using our Faces prototype as a case study, we then show how falling into these pitfalls can undermine an otherwise ordinary design process. Finally we discuss some successful systems that have largely avoided the pitfalls.

Mental Models of Information Flow

As we said earlier, avoiding our first two pitfalls—obscuring potential *and* actual information flow—can clarify the extent to which users' actions engage the system's range of privacy implications. Users can understand the consequences of their use of the system thus far, and they can predict the consequences of future use.

Illuminating disclosure contributes constructively to the user's mental model of the portrayal of her identity in the context of the system. If she has a reasonable understanding of what observers already know about her (Pitfall 2) and of what they can learn about her (Pitfall 1), she can maintain and exploit this mental model to inform the ongoing portrayal of her identity through the system.

In the context of interactive systems, the personal information a user conveys is often tightly integrated with her interaction with the system. For example, by simply browsing the web, a user generates a wealth of information that can be used in ways that directly impact her life. When interaction and disclosure are integrated thusly, an informed user's mental model of the system's operation and her mental model of her disclosures are interdependent.

This suggests an extension to Norman's canonical elucidation of the role of mental models in the design

process. According to Norman, the designer's goal is to design the system image (*i.e.*, those aspects of the implementation with which the user interacts) such that the user's mental model of the system's operation coincides with the designer's mental model of the same [29].

When we take into account the coupling of interaction and disclosure, we see that the designer's goal has expanded. She now strives to design the system image such that the user's mental models of the system's operation *and* of the portrayal of his identity through it are both accurate. As with Norman's original notion, ideally the designer's and the user's models of the system's operation will coincide. But the designer generally cannot have a model of the user's identity; that depends on the user and the context of use. Indeed, here the designer's task is not to harmonize the user's model of his information flow with her own (she likely has none), but to harmonize the user's information model with the *observer's* (Figure 1). In other words, she wants to design the system image to accurately convey a model not only of how other parties *can* observe the user's behavior through the system, but also what they *do* observe.

Generalizing this notion beyond privacy, to cooperative information flow in general, may be of further use to the CSCW community but is beyond the scope of this paper.

Opportunities for Understanding and Action

We have argued that people maintain personal privacy by *understanding* the privacy implications relevant to a situation and influencing them through intuitive *action*. When a technical system is embedded into a social process, understanding and action can occur through feedback and control. We encourage designers of privacy-affecting systems to think of feedback and control mechanisms as *opportunities* for understanding and action. They are the designer's opportunity to empower those processes, and they are the user's opportunity to practice them.

Thinking thusly can help designers reach across what Ackerman calls the *socio-technical gap*—the difference between systems' technical capabilities and their social requirements [1]—just enough to empower informed social action through the system. The challenge is to find that

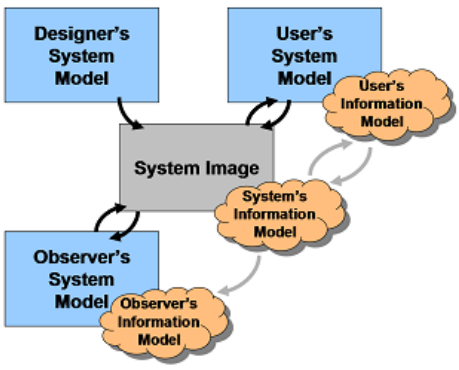

Figure 1. Building on Norman [29], designers should strive to harmonize the user's and the observer's understandings of the user's personal information disclosures.

intermediate point where carefully designed technical feedback and control translates into social understanding and action. Reaching too far can overwhelm the user. Reaching not far enough can disempower him.

Avoiding our pitfalls can help designers reach that intermediate point. Carefully designed feedback about potential (#1) and actual (#2) information flow can help users understand how their behavior and attributes are disclosed through the system. Curtailing configuration (#3), providing coarse-grained control (#4), and allowing nuanced behavior (#5) can let people act intuitively to influence that disclosure. Designs that heed these suggestions make their consequences known, they do not require great effort, and they do not inhibit existing practices. People can incorporate them meaningfully into their lexicon of personal privacy practices, with which they intuitively compose privacy as needed.

Negative Case Study: Faces

We return now to Faces—our prototypical ubicomp privacy UI—as a case study in how to fall into the pitfalls.

Pitfall 1: Obscuring Potential Flow. In trying to be a UI for managing privacy across any ubicomp system, Faces abstracted away the true capabilities of any underlying system. Users could not gauge its potential information flow because it aimed to address *all* information flow. Its scope was impractically broad and, hence, obscure.

Pitfall 2: Obscuring Actual Flow. Faces conveyed actual information flow through the user's disclosure log. The record was accessible after the relevant disclosure. While this design intends to illuminate information flow, it is unclear whether postponing notice is optimal. Embedding notice directly into the real-time experience of disclosure might foster a stronger understanding of information flow.

Pitfall 3: Configuration over Action. Faces required a considerable amount of configuration. Once configuration was done, and assuming it was done correctly (which our evaluation brings into doubt), the system required little *ad-hoc* configuration. The user simply goes about his business. Nonetheless, the sheer amount and desituated nature of configuration positions Faces squarely in this pitfall.

Pitfall 4: Lacking Coarse-grained Control. Faces avoided this pitfall somewhat by including an Override function that afforded quick transitions to alternate faces. Notably, this was not considered a central design feature. A dial to adjust disclosure precision on the fly might also be helpful.

Pitfall 5: Inhibiting Social Nuance. While Faces modeled the nuance of Goffman's identity management theory, it appeared to hinder the actual identity management practice by requiring the user to maintain virtual representations of his fragmented identities *in addition to* manifesting them naturally through action. A simple *precision dial* might free the user from this pre-creation and maintenance process.

Positive Case Study: IM and Mobile Phones

Interestingly, two systems that largely avoid our pitfalls—mobile phones and instant messaging—are primarily communication media. Disclosure is a central aspect of their design. Each makes clear the *scope* and *flow* of disclosed information, through Caller ID, Buddy Lists, and reflecting the user's online presence. Each requires minimal *configuration* for maintaining privacy; conveying and withholding information are built in to the primary actions one makes through the system. Each provides a *coarse-grained control* for halting and resuming flow, through power buttons, application exit, invisible mode, and ringer volume. And each supports social *nuance* through plausible deniability and opportunities for ambiguous disclosure.

The design of communication media could serve as a model for deigning other privacy-affecting systems. Disclosure *is* essentially communication. Systems that affect privacy but are not positioned as communication media (*e.g.*, e-commerce, context-aware systems) do nonetheless communicate personal information to observers. Exposing and addressing these disclosure media as communication media can liberate designs to leverage users' intuitive privacy maintenance skills.

CONCLUSION

In this paper we described five common pitfalls that designs of privacy-affecting systems often fall into. These pitfalls include obscuring potential information flow, obscuring actual flow, emphasizing configuration over action, lacking coarse-grained control, and inhibiting social nuance. We analyzed these pitfalls and provided several examples of systems that fall into or manage to avoid them, including Faces, our UI prototype for managing ubicomp privacy.

We showed that, by avoiding our pitfalls, systems can create opportunities for users to create and maintain personal privacy though *understanding* the privacy implications relevant to a situation and influencing them through intuitive social *action*. Technical feedback and control mechanisms that enable these human processes are the key to empowering people to maintain personal privacy.

REFERENCES

1. Ackerman, M.S. The Intellectual Challenge of CSCW: The Gap Between Social Requirements and Technical Feasibility. Human-Computer Interaction, 15 (2/3). 181-203. 2000.
2. Adams, A. Multimedia Information Changes the Whole Privacy Ballgame. Proc. Computers, Freedom, and Privacy. 2000. 25-32.
3. Adams, A. and Sasse, M.A. Taming the Wolf in Sheep's Clothing: Privacy in Multimedia Communications. Proc. ACM Multimedia '99. 101-107.
4. Beckwith, R. Designing for Ubiquity: The Perception of Privacy IEEE Pervasive 2(2). 2003. 40-46.
5. Bellotti, V., *et al*. Making sense of sensing systems: five questions for designers and researchers. Proc. CHI 2002. 415-422.
6. Bellotti, V. and Sellen, A., Design for Privacy in Ubiquitous Computing Environments. Proc. ECSCW'93. 77-92.
7. boyd, d. Faceted Id/Entity: Managing representation in a digital world, MS Thesis, MIT Media Lab, 2002.
8. boyd, d., Reflections on Friendster, Trust and Intimacy. Proc. Workshop on Intimate Ubiquitous Computing, Ubicomp 2003.
9. Cadiz, J. and Gupta, A. Privacy Interfaces for Collaboration, Technical Report MSR-TR-2001-82, Microsoft Corp., 2001.
10. Cranor, L., *et al*. Beyond Concern: Understanding Net Users' Attitudes About Online Privacy. in Vogelsang, I. and Compaine, B.M. eds. The Internet Upheaval: Raising Questions, Seeking Answers in Communications Policy, MIT Press. 2000, 47-70.
11. Dey, A.K., *et al*. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. Human-Computer Interaction, 16 (2-4). 2001. 97-166.
12. Foucault, M. Discipline and Punish. Vintage, New York, 1977.
13. Friedman, B., *et al*. Informed Consent in the Mozilla Browser: Implementing Value-Sensitive Design. Proc. Hawaii International Conference on System Sciences, 2002.
14. Goffman, E. The Presentation of Self in Everyday Life. Doubleday, New York, NY, 1956.
15. Good, N.S. and Krekelberg, A. Usability and privacy: a study of Kazaa P2P file-sharing. Proc. CHI 2003. 137-144.
16. Green, N., Lachoee, H. and Wakeford, N., Rethinking Queer Communications: Mobile Phones and beyond. Proc. Sexualities, Medias and Technologies Conference. 2001.
17. Harper, R.H.R., *et al*. Locating Systems at Work: Implications for the Development of Active Badge Applications. Interacting with Computers, 4 (3). 1992. 343-363.
18. Jancke, G., *et al*. Linking public spaces: technical and social issues. Proc. CHI 2001. 530-537.
19. Jendricke, U. and Markotten, D.G.t., Usability meets Security – The Identity-Manager as your Personal Security Assistant for the Internet. Proc. 16th Annual Computer Security Applications Conference. 2000.
20. Jiang, X., *et al*. Approximate Information Flows: Socially-based Modeling of Privacy in Ubiquitous Computing. Proc. UBICOMP 2002. 176-193.
21. Kaasinen, E. User needs for location-aware mobile services. Personal and Ubiquitous Computing, 7 (1). 2003. 70-79.
22. Langheinrich, M., Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems. Proc. UBICOMP 2001. 273-291.
23. Lederer, S., *et al*. Towards a Deconstruction of the Privacy Space. To appear at workshop on Ubicomp Communities: Privacy as Boundary Negotiation, at UBICOMP 2003.
24. Lederer, S., *et al*. Who wants to know what when? Privacy preference determinants in ubiquitous computing. Proc. CHI '03 extended abstracts, 724--725.
25. Lederer, S., *et al*. Managing Personal Information Disclosure in Ubiquitous Computing Environments, Technical Report CSD-03-1257, UC Berkeley, Berkeley, CA, 2003.
26. Mackay, W.E., Triggers and barriers to customizing software. Proc. CHI '99. 153-160.
27. Millett, L.I., *et al*. Cookies and Web browser design: toward realizing informed consent online. Proc. CHI 2001. 46-52.
28. Nardi, B.A., *et al*. Interaction and Outeraction: Instant Messaging in Action. Proc. CSCW 2000. 79-88.
29. Norman, D.A. The Design of Everyday Things. Basic Books, New York, NY, 1988.
30. Palen, L., Social, Individual & Technological Issues for Groupware Calendar Systems. Proc. CHI' 99. 17-24.
31. Palen, L. and Dourish, P., Unpacking "privacy" for a networked world. Proc. CHI 2003. 129-136.
32. Reang, P. Dozens of nurses in Castro Valley balk at wearing locators *Mercury News*, San Jose, CA, 2002.
33. Siewiorek, D., *et al*. SenSay: A Context-Aware Mobile Phone. To appear in Proc. International Symposium on Wearable Computers. 2003.

34. Taylor, H. Most People Are "Privacy Pragmatists" Who, While Concerned about Privacy, Will Sometimes Trade It Off for Other Benefits, The Harris Poll, 2003.
35. Turow, J. Americans and Online Privacy: The System is Broken, Annenberg Public Policy Center, University of Pennsylvania, 2003.
36. Weiser, M. The Computer for the Twenty-First Century Scientific American 265(3). 1991. 94-104.
37. Westin, A. *Privacy and Freedom*. Atheneum, New York, 1967.
38. Whitten, A. and Tygar, J.D. Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0. Proc. 8th USENIX Security Symposium. 1999.
39. Woodruff, A. and Aoki, P.M. How Push-to-Talk Makes Talk Less Pushy. To appear in Proc. GROUP '03.